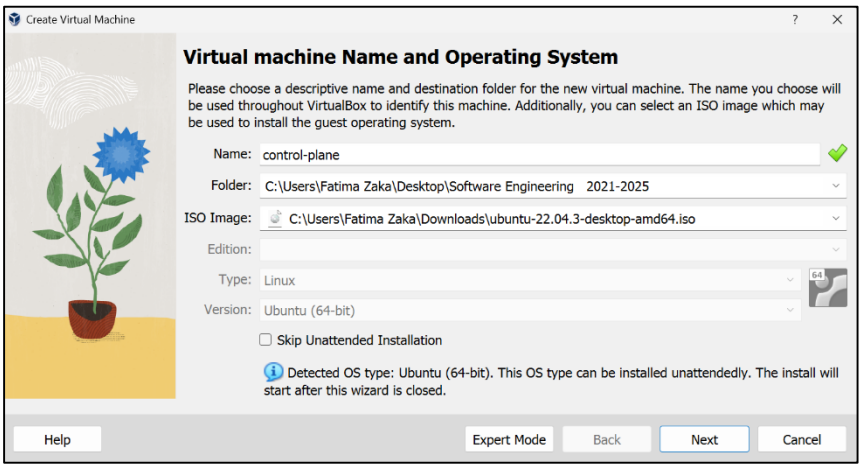
CLOUD COMPUTING LAB PROJECT

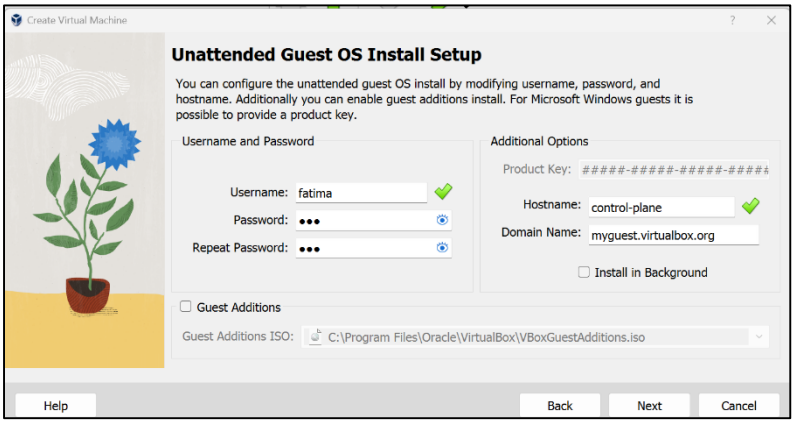
SUBMITTED BY:

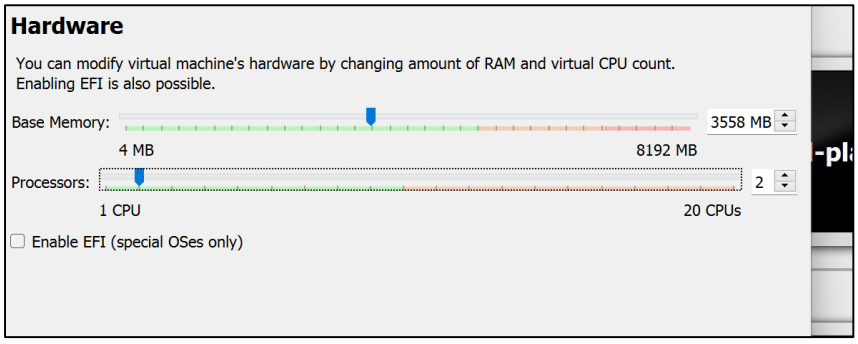
- LAIBA SOHAIL(2021-BSE-016)
- TANZEELA ASGHAR(2021-BSE-032)
- BISMA ALI (2021-BSE-008)

“DEPLOYING JAVA APPLICATIONS WITH DOCKER AND KUBERNETES”

INSTALLING UBUNTU:





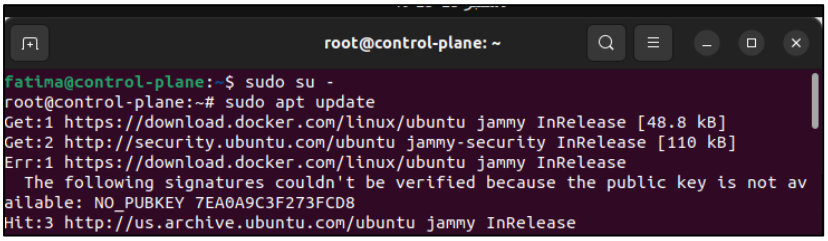


To deploy java application on Kubernetes:

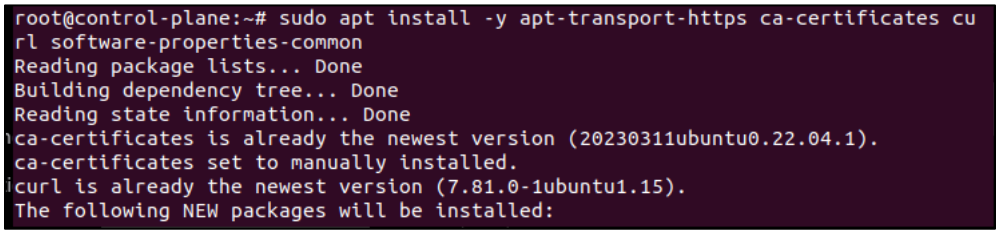
- All code in git repository -sample project in this video-
- From github checking out the code
- The code is base code; we don't have the target file like .jar or .var
- We need to build code to cinvert it to .jar file
- We will use build maven to convert it in jar file
- NOW u have build application (jar file)
- Then create docker image
- We will have the docker file to copy the jar file in docker image
- When we run docker image in container , we can run the application

A. Install Docker on Virtual Machine

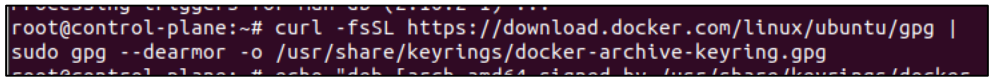
1. Update packages



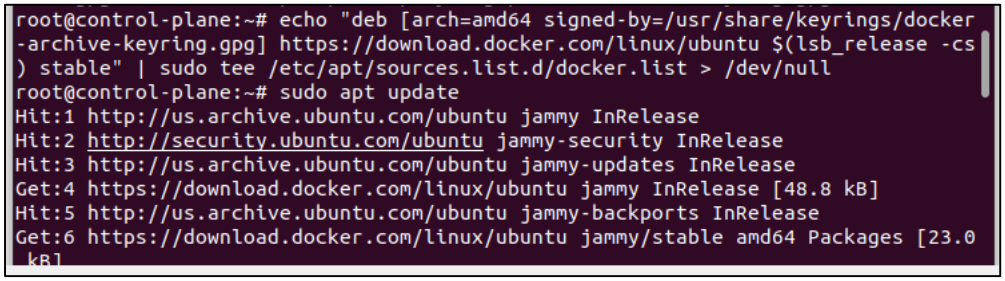
2. Install required dependencies



3. Add the docker GPG key



4. Set up the stable Docker repository



5. Update the package list again

```
root@control-plane:~# sudo apt update
Hit:1 http://us.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu jammy-updates InRelease
Get:4 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Hit:5 http://us.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:6 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [23.0
 kB]
Fetched 71.8 kB in 2s (40.0 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
170 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

6. Install Docker Engine

```
root@control-plane:~# sudo apt install -y docker-ce docker-ce-cli containerd.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  docker-buildx-plugin docker-ce-rootless-extras docker-compose-plugin git
  git-man liberror-perl libslirp0 pigz slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvinit
  git-doc git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli
  docker-ce-rootless-extras docker-compose-plugin git git-man liberror-perl
  libslirp0 pigz slirp4netns
0 upgraded, 12 newly installed, 0 to remove and 170 not upgraded.
Need to get 119 MB of archives.
After this operation, 432 MB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu jammy/universe amd64 pigz amd64 2.6-1
[63.6 kB]
Get:2 https://download.docker.com/linux/ubuntu jammy/stable amd64 containerd.io
amd64 1.6.26-1 [29.5 MB]
Get:3 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 liberror-perl all 0.1
```

7. Verify Docker installation

```
root@control-plane:~# sudo docker --version
Docker version 24.0.7, build afdd53b
root@control-plane:~# S
```

8. Enable docker

```
root@control-plane:~# sudo systemctl enable docker
Synchronizing state of docker.service with SysV service script with /lib/system
d/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable docker
```

9. Check docker status

```
root@control-plane:~# sudo systemctl status  docker
* docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset
>
     Active: active (running) since Mon 2023-12-25 19:23:03 PKT; 12min ago
TriggeredBy: * docker.socket
       Docs: https://docs.docker.com
   Main PID: 7364 (dockerd)
      Tasks: 9
     Memory: 28.1M
        CPU: 1.037s
     CGroup: /system.slice/docker.service
             `-7364 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/cont
>

19:23:01 25 بر <85>ش control-plane systemd[1]: Starting Docker Application>
19:23:01 25 بر <85>ش control-plane dockerd[7364]: time="2023-12-25T19:23:0>
19:23:01 25 بر <85>ش control-plane dockerd[7364]: time="2023-12-25T19:23:0>
19:23:02 25 بر <85>ش control-plane dockerd[7364]: time="2023-12-25T19:23:0>
19:23:03 25 بر <85>ش control-plane dockerd[7364]: time="2023-12-25T19:23:0>
19:23:03 25 بر <85>ش control-plane dockerd[7364]: time="2023-12-25T19:23:0>
19:23:03 25 بر <85>ش control-plane dockerd[7364]: time="2023-12-25T19:23:0>
```

B. **INSTALL MINIKUBE**

```
fatima@control-plane: ~

fatima@control-plane:~$ curl -LO https://storage.googleapis.com/minikube/release
s/latest/minikube_latest_amd64.deb
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 28.9M  100 28.9M    0     0  2056k      0  0:00:14  0:00:14 --:--:-- 2401k
fatima@control-plane:~$ sudo dpkg -i minikube_latest_amd64.deb
(Reading database ... 206488 files and directories currently installed.)
Preparing to unpack minikube_latest_amd64.deb ...
Unpacking minikube (1.32.0-0) over (1.32.0-0) ...
Setting up minikube (1.32.0-0) ...
fatima@control-plane:~$ minikube start
😄  minikube v1.32.0 on Ubuntu 22.04 (vbox/amd64)
✨  Automatically selected the docker driver. Other choices: none, ssh
📌  Using Docker driver with root privileges
👍  Starting control plane node minikube in cluster minikube
🚜  Pulling base image ...
💾  Downloading Kubernetes v1.28.3 preload ...
    > preloaded-images-k8s-v18-v1...:  403.35 MiB / 403.35 MiB  100.00% 387.91
    > gcr.io/k8s-minikube/kicbase...:  421.78 MiB / 453.90 MiB  92.92% 555.44 K
    > gcr.io/k8s-minikube/kicbase...:  453.90 MiB / 453.90 MiB  100.00% 400.79
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...
🐳  Preparing Kubernetes v1.28.3 on Docker 24.0.7 ...
    ▪ Generating certificates and keys ...
    ▪ Booting up control plane ...
    ▪ Configuring RBAC rules ...
🔗  Configuring bridge CNI (Container Networking Interface) ...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🔎  Verifying Kubernetes components...
🌟  Enabled addons: storage-provisioner, default-storageclass
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" name
space by default
```

```
fatima@control-plane:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
```

```
fatima@control-plane:~$ sudo usermod -aG docker $USER
[sudo] password for fatima:
fatima@control-plane:~$ newgrp docker
fatima@control-plane:~$
```

C. INSTALL KUBERNETES AND KUBERNETES TOOLS

```
fatima@control-plane:~$ sudo apt-get update
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:2 http://us.archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://us.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:4 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:6 http://us.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:5 https://packages.cloud.google.com/apt kubernetes-xenial InRelease
Fetched 229 kB in 5s (47.8 kB/s)
Reading package lists... Done
fatima@control-plane:~$ sudo apt-get install -y kubectl kubeadm kubelet
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
kubeadm is already the newest version (1.28.2-00).
kubectl is already the newest version (1.28.2-00).
kubelet is already the newest version (1.28.2-00).
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
```

```
fatima@control-plane:~$ sudo apt-get update && sudo apt-get install -y apt-trans
port-https curl
Hit:1 http://us.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:6 https://packages.cloud.google.com/apt kubernetes-xenial InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
curl is already the newest version (7.81.0-1ubuntu1.15).
apt-transport-https is already the newest version (2.4.11).
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
```

```
fatima@control-plane:~$ sudo apt-mark hold kubelet kubeadm kubectl
kubelet was already set on hold.
kubeadm was already set on hold.
kubectl was already set on hold.
```

D. SETTING UP KUBERNETES

```
fatima@control-plane:~$ kubectl get nodes
NAME       STATUS   ROLES           AGE   VERSION
minikube   Ready    control-plane   18m   v1.28.3
```

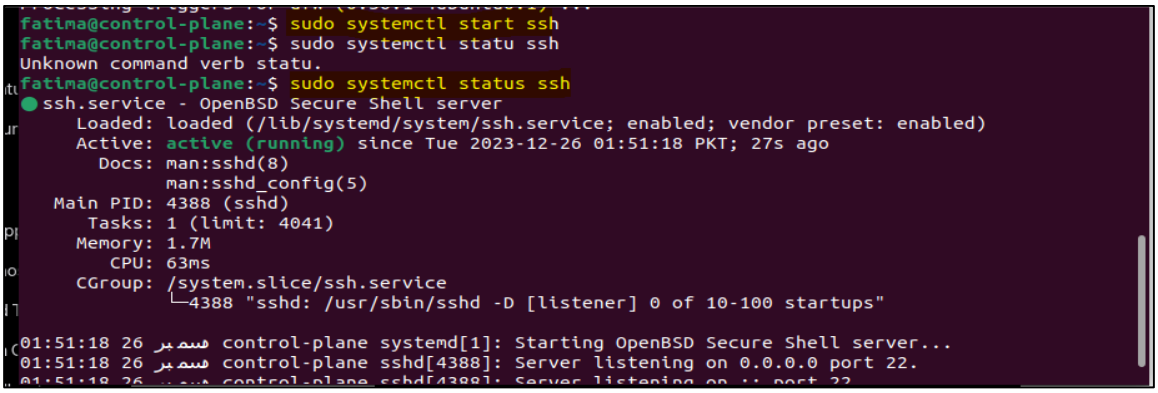
```
fatima@control-plane:~$ kubectl cluster-info
Kubernetes control plane is running at https://192.168.58.2:8443
CoreDNS is running at https://192.168.58.2:8443/api/v1/namespaces/kube-system/services/kube-dns:dns
/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
fatima@control-plane:~$
```

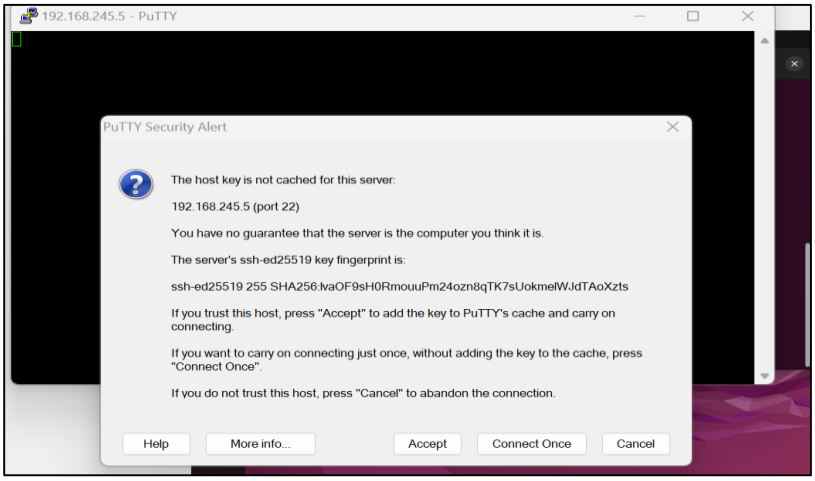
E. CREATE DOCKER ID

```
root@UBUNTUU:~# docker login
Log in with your Docker ID or email address to push and pull images from Docker
Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to crea
te one.
You can log in with your password or a Personal Access Token (PAT). Using a limi
ted-scope PAT grants better security and is required for organizations using SSO
. Learn more at https://docs.docker.com/go/access-tokens/

Username: laiba280
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

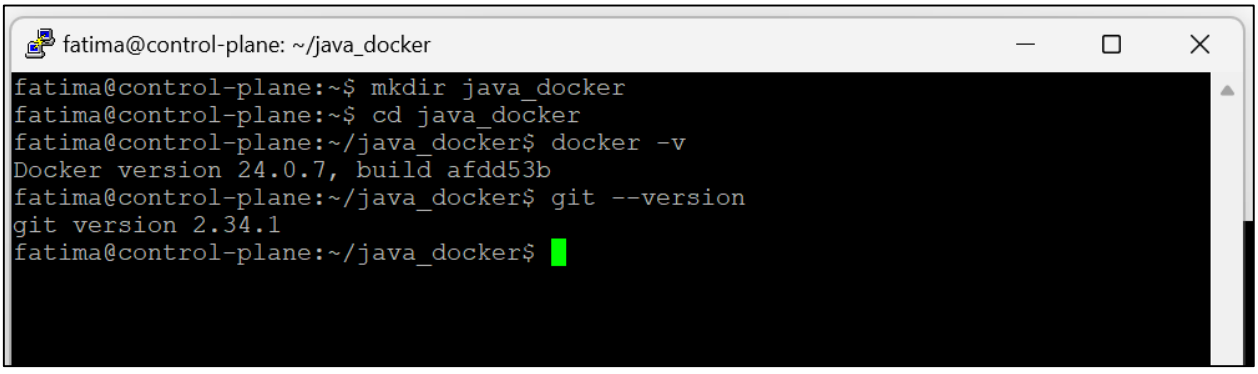
F. INSTALL OPENSSH AND CONNECTION WITH PUTTY

```
fatima@control-plane: ~
fatima@control-plane:~$ sudo apt-get install -y openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere ssh-askpass
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 4 newly installed, 0 to remove and 5 not upgraded.
Need to get 752 kB of archives.
After this operation, 6,050 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu jammy-updates/main amd64 openssh-sftp-server amd64 1:8.9p
```





G. CREATE DIRECTORY AND CHECK VERSIONSSS



H. INSTALL MAVEN’

```
fatima@control-plane:~/java_docker$ sudo apt install maven
[sudo] password for fatima:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java default-jre-headless java-common libaopalliance-java
  libapache-pom-java libatinject-jsr330-api-java libcdi-api-java
  libcommons-cli-java libcommons-io-java libcommons-lang3-java
  libcommons-parent-java libgeronimo-annotation-1.3-spec-java
  libgeronimo-interceptor-3.0-spec-java libguava-java libguice-java
  libhawtjni-runtime-java libjansi-java libjansi-native-java libjsr305-java
  libmaven-parent-java libmaven-resolver-java libmaven-shared-utils-java
  libmaven3-core-java libplexus-cipher-java libplexus-classworlds-java
  libplexus-component-annotations-java libplexus-interpolation-java
  libplexus-sec-dispatcher-java libplexus-utils2-java libsisu-inject-java
  libsisu-plexus-java libslf4j-java libwagon-file-java
```

```
Adding debian:Trustwave_Global_Certification_Authority.pem
Adding debian:ISRG_Root_X1.pem
Adding debian:Trustwave_Global_ECC_P256_Certification_Authority.pem
Adding debian:Starfield_Root_Certificate_Authority_-_G2.pem
Adding debian:GTS_Root_R2.pem
Adding debian:vTrus_Root_CA.pem
Adding debian:UCA_Global_G2_Root.pem
Adding debian:emSign_ECC_Root_CA_-_G3.pem
done.
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for ca-certificates (20230311ubuntu0.22.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...

done.
done.
```

➔ Export

```
fatima@control-plane:~/java_docker$ export MAVEN_HOME=/usr/share/maven
fatima@control-plane:~/java_docker$ echo $MAVEN_HOME
/usr/share/maven
fatima@control-plane:~/java_docker$
```

```
fatima@control-plane:~/java_docker$ mvn --version
Apache Maven 3.6.3
Maven home: /usr/share/maven
Java version: 11.0.21, vendor: Ubuntu, runtime: /usr/lib/jvm/java-11-openjdk-amd
64
Default locale: en_CA, platform encoding: UTF-8
OS name: "linux", version: "6.2.0-39-generic", arch: "amd64", family: "unix"
fatima@control-plane:~/java_docker$
```

MAVEN_HOME as an environmental variabl

➔ Now maven is set up.

I. NOW, BUILD APPLICATION

➔ Clone the git repository link given below the video

```
fatima@control-plane:~/java_docker$ git clone https://github.com/shazforiot/dock
er-java-kubernetes-project.git
Cloning into 'docker-java-kubernetes-project'...
remote: Enumerating objects: 129, done.
remote: Counting objects: 100% (28/28), done.
remote: Compressing objects: 100% (19/19), done.
remote: Total 129 (delta 15), reused 9 (delta 9), pack-reused 101
Receiving objects: 100% (129/129), 22.21 KiB | 110.00 KiB/s, done.
Resolving deltas: 100% (20/20), done.
fatima@control-plane:~/java_docker$ ls
docker-java-kubernetes-project
fatima@control-plane:~/java_docker$
```

➔ Check the folders

```
fatima@control-plane:~/java_docker$ cd docker-java-kubernetes-project
fatima@control-plane:~/java_docker/docker-java-kubernetes-project$ ls
kubernetes  productcatalogue  README.md  shopfront  stockmanager
```

➔ Goto shopfront

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project$ cd shopfront/
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$ ls
Dockerfile  pom.xml  src
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$
```

➔ There is no target file(.jar) file.

J. CREATING JAR FILE

```
root@user-VirtualBox:~/java_docker/docker-java-kubernetes-project/shopfront# mvn clean install
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/lib/g
od java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.ProtectionDomain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Scanning for projects...
```

```
1.6.jar (233 kB at 792 kB/s)
[INFO] Installing /home/fatima/java_docker/docker-java-kubernetes-project/shopfront/target/shopfront-0.0.1-S
NAPSHOT.jar to /home/fatima/.m2/repository/uk/co/danielbryant/djshopping/shopfront/0.0.1-SNAPSHOT/shopfront-
0.0.1-SNAPSHOT.jar
[INFO] Installing /home/fatima/java_docker/docker-java-kubernetes-project/shopfront/pom.xml to /home/fatima/
.m2/repository/uk/co/danielbryant/djshopping/shopfront/0.0.1-SNAPSHOT/shopfront-0.0.1-SNAPSHOT.pom
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  02:11 min
[INFO] Finished at: 2023-12-26T02:26:23+05:00
[INFO] ------------------------------------------------------------------------
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$
```

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$ ls
Dockerfile  pom.xml  src  target
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$
```

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$ cd target
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront/target$ ls
classes                 maven-archiver                 shopfront-0.0.1-SNAPSHOT.jar.original
generated-sources       maven-status                   surefire-reports
generated-test-sources  shopfront-0.0.1-SNAPSHOT.jar   test-classes
```

➔ Now use this jar file to build docker image.

K. BUILD DOCKER IMAGE USING THE DOCKER ID

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$ docker build -t fatimazz/shopf
ront:latest .
[+] Building 0.6s (2/3)                                                                          docker:default
 => [internal] load .dockerignore                                                                          0.2s
 => => transferring context: 2B                                                                            0.0s
 => [internal] load build definition from Dockerfile                                                       0.2s
 => => transferring dockerfile: 198B                                                                       0.0s
 => [internal] load metadata for docker.io/library/openjdk:8-jre                                           0.3s
```

```
=> => extracting sha256:73d77b4774a96dfa09076212d5170e977d153ceab60c1ec4312a8f436b91371c
=> [2/2] ADD target/shopfront-0.0.1-SNAPSHOT.jar app.jar
=> exporting to image
=> => exporting layers
=> => writing image sha256:4409afe0d13234bdbcb046da767d88dae578a0cd6f77daea939b103c5a94591f
=> => naming to docker.io/fatimazz/shopfront:latest
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$
```

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$ docker images
REPOSITORY                    TAG       IMAGE ID       CREATED          SIZE
fatimazz/shopfront            latest    4409afe0d132   2 minutes ago    320MB
gcr.io/k8s-minikube/kicbase   v0.0.42   dbc648475405   7 weeks ago      1.2GB
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$
```

L.

➔ DO THE SAME THING FOR OTHER MICROSERVICE AS WELL

1. Productcatalogue

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/shopfront$ cd ..
fatima@control-plane:~/java_docker/docker-java-kubernetes-project$ cd productcatalogue
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/productcatalogue$ ls
Dockerfile  pom.xml  product-catalogue.yml  src
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/productcatalogue$
```

➔ create the jar file required to build docker image.

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/productcatalogue$ mvn clean install
[INFO] Scanning for projects...
[WARNING]
[WARNING] Some problems were encountered while building the effective model for uk.co.danielbryant.djshoppi
ng:productcatalogue:jar:0.0.1-SNAPSHOT
[WARNING] 'build.plugins.plugin.version' for org.apache.maven.plugins:maven-compiler-plugin is missing. @ l
ine 33, column 21
[WARNING]
[WARNING] It is highly recommended to fix these problems because they threaten the stability of your build.
[WARNING]
[WARNING] For this reason, future Maven versions might no longer support building such malformed projects.
[WARNING]
[INFO]
[INFO] -----------< uk.co.danielbryant.djshopping:productcatalogue >-----------
[INFO] Building productcatalogue 0.0.1-SNAPSHOT
[INFO] --------------------------------[ jar ]---------------------------------
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/
2.5/maven-clean-plugin-2.5.pom
```

```
productcatalogue 0.0.1-SNAPSHOT.pom
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  50.100 s
[INFO] Finished at: 2023-12-26T02:41:11+05:00
[INFO] ------------------------------------------------------------------------
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/productcatalogue$ ls
Dockerfile  pom.xml  product-catalogue.yml  src  target
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/productcatalogue$
```

➔ Now, build the docker image.

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/productcatalogue$ docker build -t fatimaz
z/productcatalogue:latest .
[+] Building 0.7s (2/3)                                                                          docker:default
 => [internal] load build definition from Dockerfile                                                      0.1s
 => => transferring dockerfile: 273B                                                                      0.0s
 => [internal] load .dockerignore                                                                         0.1s
 => => transferring context: 2B                                                                           0.0s
 => [internal] load metadata for docker.io/library/openjdk:8-jre                                          0.5s
```

➔ Docker image is created.

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/productcatalogue$ docker images
REPOSITORY                    TAG       IMAGE ID       CREATED          SIZE
fatimazz/productcatalogue     latest    832ea4a6ac12   40 seconds ago   291MB
fatimazz/shopfront            latest    4409afe0d132   9 minutes ago    320MB
gcr.io/k8s-minikube/kicbase   v0.0.42   dbc648475405   7 weeks ago      1.2GB
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/productcatalogue$
```

2. Stockmanager file

➔ Change the directory to stockmanager and create a jar file first

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/productcatalogue$ cd ..
fatima@control-plane:~/java_docker/docker-java-kubernetes-project$ cd stockmanager
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$ ls
Dockerfile  pom.xml  src
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$ mvn clean install
[INFO] Scanning for projects...
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-starter
-parent/2.3.7.RELEASE/spring-boot-starter-parent-2.3.7.RELEASE.pom
```

```
ContextBootstrapper=true, server.port=0}

  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::        (v2.3.7.RELEASE)

2023-12-26 02:46:55.083  INFO 6812 --- [           main] u.c.d.d.s.ShopfrontApplicationTests      : Startin
g ShopfrontApplicationTests on control-plane with PID 6812 (started by fatima in /home/fatima/java_docker/d
ocker-java-kubernetes-project/stockmanager)
2023-12-26 02:46:55.100  INFO 6812 --- [           main] u.c.d.d.s.ShopfrontApplicationTests      : No acti
ve profile set, falling back to default profiles: default
2023-12-26 02:46:57.594  INFO 6812 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstr
apping Spring Data JPA repositories in DEFAULT mode.
2023-12-26 02:46:57.756  INFO 6812 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Finishe
d Spring Data repository scanning in 140ms. Found 1 JPA repository interfaces.
```

```
m
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  01:48 min
[INFO] Finished at: 2023-12-26T02:47:35+05:00
[INFO] ------------------------------------------------------------------------
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$ ls
build  Dockerfile  pom.xml  src  target
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$
```

➔ Build docker image

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$ docker build -t fatimazz/st
ockmanager:latest .
[+] Building 0.8s (2/3)                                                                    docker:default
 => [internal] load build definition from Dockerfile                                                0.1s
 => => transferring dockerfile: 201B                                                                0.0s
 => [internal] load .dockerignore                                                                   0.2s
 => => transferring context: 2B                                                                     0.0s
 => [internal] load metadata for docker.io/library/openjdk:8-jre                                    0.6s
```

M. NOW PUSH ALL THE DOCKER IMAGED TO DOCKER HUB

➔ LOGIN TO DOCKER

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$ docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Do
cker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better
 security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-toke
ns/

Username: fatimazz
Password:
WARNING! Your password will be stored unencrypted in /home/fatima/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$
```

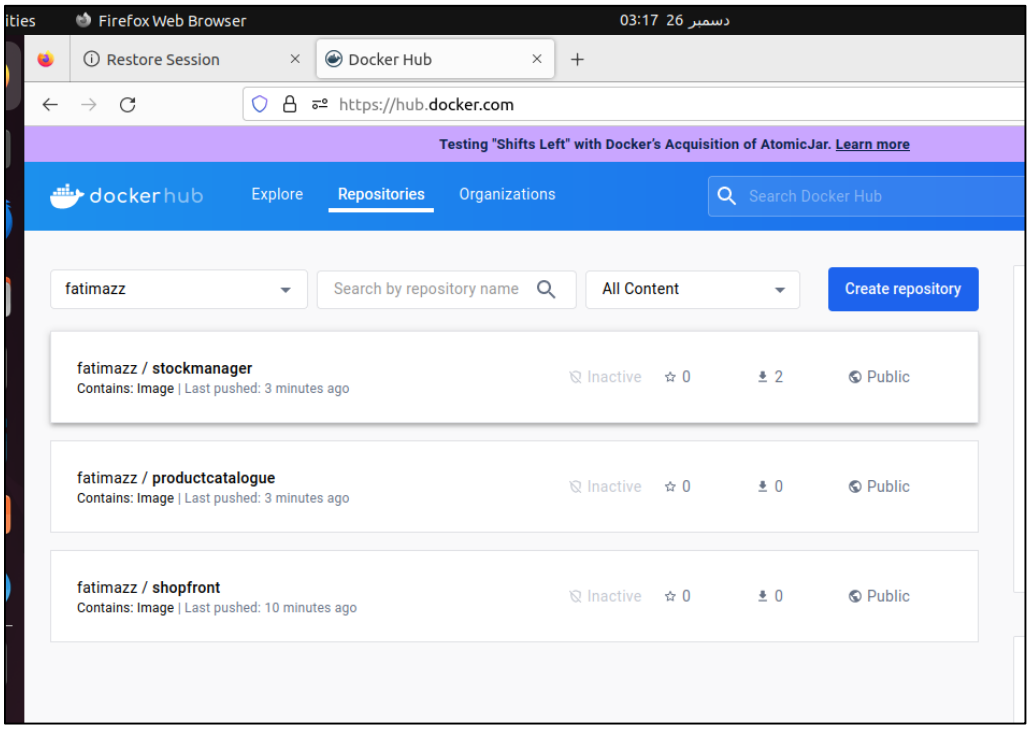
➔ PUSH THE THREE IMAGES INTO DOCKER HUB

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$ docker push fatimazz/stockm
anager
Using default tag: latest
The push refers to repository [docker.io/fatimazz/stockmanager]
0e4fde77ceea: Preparing
1aaddf64804f: Preparing
990c5138f5d1: Preparing
5c384ea5f752: Preparing
293d5db30c9f: Preparing
03127cdb479b: Waiting
9c742cd6c7a5: Waiting
```

```
03127cdb479b: Mounted from library/openjdk
9c742cd6c7a5: Mounted from library/openjdk
latest: digest: sha256:894855996537d5d61e661e0988a2ef648bad451d3f5379fdd7088c15ff981d6c size: 1794
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$
```

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$ docker push fatimazz/produc
tcatalogue:latest
The push refers to repository [docker.io/fatimazz/productcatalogue]
e92565c67805: Pushing  2.048kB
d5e949500459: Preparing
1aaddf64804f: Preparing
990c5138f5d1: Preparing
5c384ea5f752: Preparing
293d5db30c9f: Waiting
03127cdb479b: Waiting
9c742cd6c7a5: Waiting
```

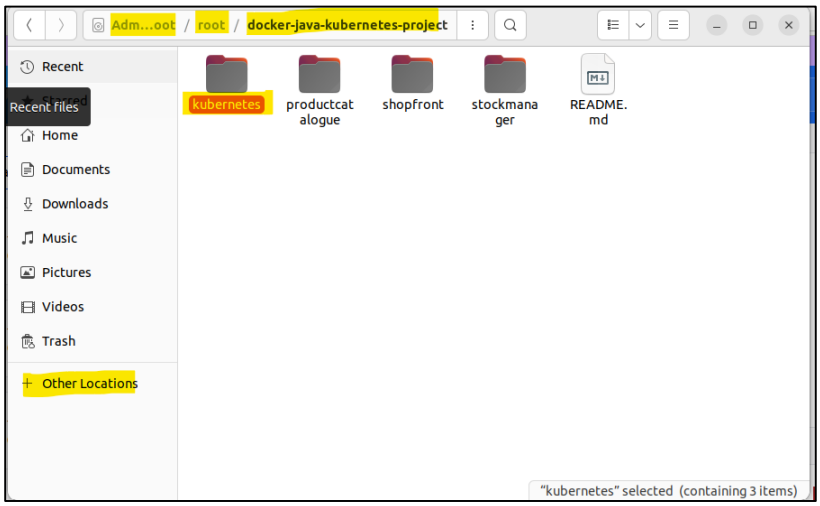
➔ **All** the images will appear in your docker hub account in repositories section

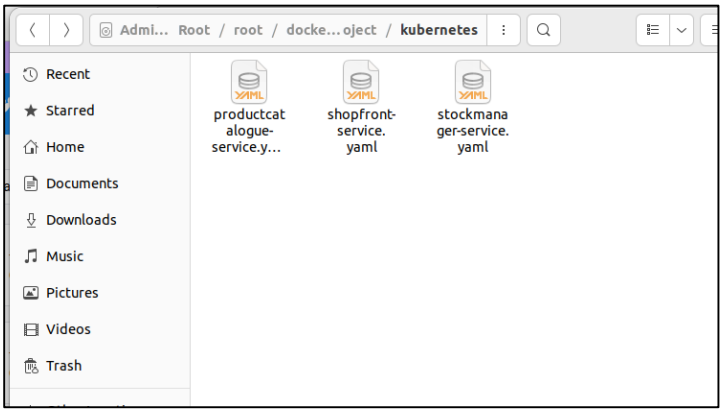


N. EDIT THIS WITH YOUR DOCKER ID IN ALL yaml FILES

```
- name: shopfront
  image: fatimazz/shopfront:latest
  ports:
  - containerPort: 8010
  livenessProbe:
    httpGet:
      path: /health
      port: 8010
    initialDelaySeconds: 30
    timeoutSeconds: 1
```

O. EDIT THE .YAML FILES PRESENT IN ROOT (IN KUBERNETES FOLDER)





➔ CHANGE THE IMAGE NAME TO YOUR IMAGE NAME

```
30        app: productcatalogue
31    spec:
32      containers:
33      - name: productcatalogue
34        image: thetips4you/productcatalogue:latest
35        ports:
36        - containerPort: 8020
37        livenessProbe:
```

```
31    spec:
32      containers:
33      - name: productcatalogue
34        image: fatimazz/productcatalogue:latest
35        ports:
36        - containerPort: 8020
37        livenessProbe:
```

P. **APPLY YAML FILES IN KUBERNETES FOLDER FOR SERVICES AND DEPLOYMENT**

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/stockmanager$ cd ..
fatima@control-plane:~/java_docker/docker-java-kubernetes-project$ cd kubernetes
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes$ ls
productcatalogue-service.yaml  shopfront-service.yaml  stockmanager-service.yaml
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes$
```

```
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes$ ls
productcatalogue-service.yaml  shopfront-service.yaml  stockmanager-service.yaml
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes$ kubectl apply -f productcatalogue-service.yaml
service/productcatalogue created
deployment.apps/productcatalogue created
fatima@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes$ kubectl apply -f shopfront-service.yaml
service/shopfront created
deployment.apps/shopfront created
```

```
root@control-plane:~/java_docker/docker-java-kubernetes-project# cd kubernetes
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes# ls
productcatalogue-service.yaml  stockmanager-service.yaml
shopfront-service.yaml
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes# kube
ctl apply -f productcatalogue-service.yaml
service/productcatalogue unchanged
deployment.apps/productcatalogue unchanged
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes# kube
ctl apply -f stockmanager-service.yaml
service/stockmanager unchanged
deployment.apps/stockmanager unchanged
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes# kube
ctl apply -f shopfront-service.yaml
service/shopfront unchanged
deployment.apps/shopfront unchanged
```

Pods

```
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes# kube
ctl get pods
NAME                              READY   STATUS              RESTARTS   AGE
productcatalogue-78cbc478f6-kl5tq   1/1     Running             0          18m
shopfront-5b94db65cb-g9hhj          0/1     ContainerCreating   0          18m
stockmanager-645c66fc68-fxhjr       1/1     Running             0          18m
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes# kube
ctl get pods
NAME                              READY   STATUS    RESTARTS      AGE
productcatalogue-78cbc478f6-kl5tq   1/1     Running   0             20m
shopfront-5b94db65cb-g9hhj          1/1     Running   1 (42s ago)   20m
stockmanager-645c66fc68-fxhjr       1/1     Running   2 (27s ago)   20m
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes#
```

Check deployment

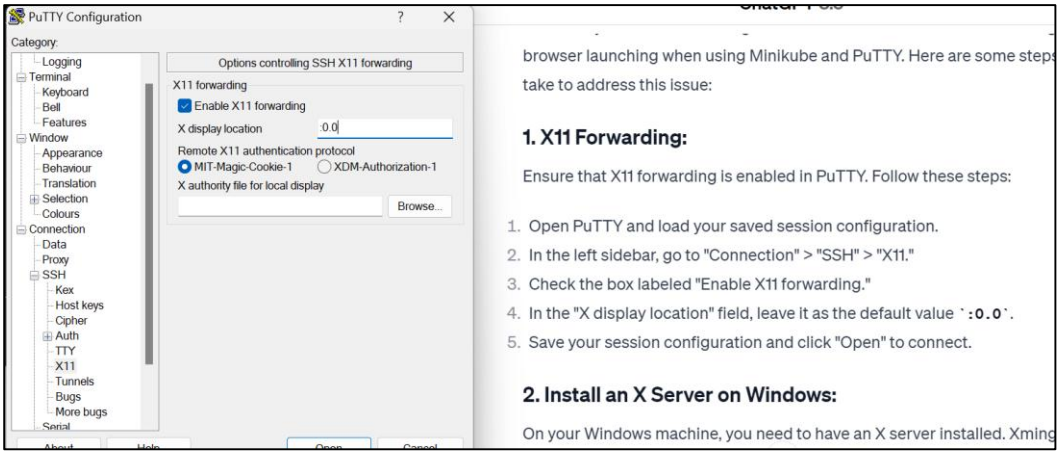
```
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes# kube
ctl get deployment
NAME              READY   UP-TO-DATE   AVAILABLE   AGE
productcatalogue  1/1     1            1           21m
shopfront         1/1     1            1           20m
stockmanager      1/1     1            1           21m
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes#
```

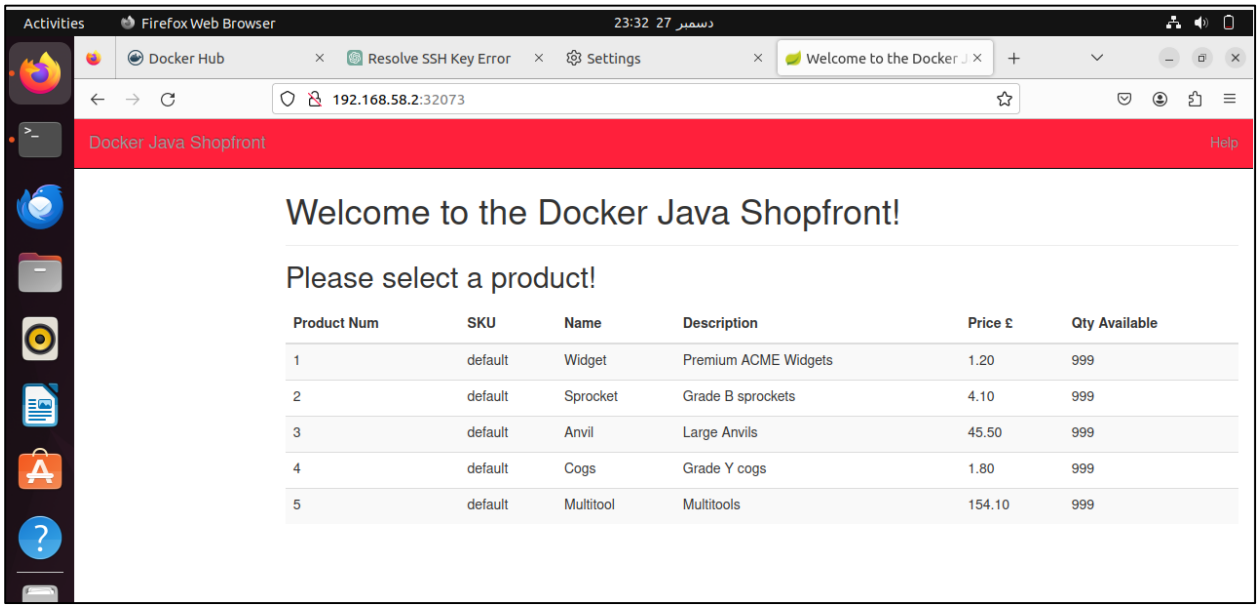
Check services

```
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes# kube
ctl get svc
NAME              TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)          A
GE
kubernetes        ClusterIP   10.96.0.1        <none>        443/TCP          5
h47m
productcatalogue  NodePort    10.109.195.113   <none>        8020:30278/TCP   2
2m
shopfront         NodePort    10.107.19.113    <none>        8010:32073/TCP   2
1m
stockmanager      NodePort    10.100.193.6     <none>        8030:32341/TCP   2
1m
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes#
```

➔ Shopfront is our main frontend

Access applicationn through web browser






```
root@control-plane:~/java_docker/docker-java-kubernetes-project/kubernetes# minikub
e service shopfront
|-----------|-----------|-------------|--------------------------|
| NAMESPACE |   NAME    | TARGET PORT |            URL           |
|-----------|-----------|-------------|--------------------------|
| default   | shopfront | http/8010   | http://192.168.58.2:32073 |
|-----------|-----------|-------------|--------------------------|
```

Q. Install Kubernetes Tools (kubeadm, kubelet, kubectl)

1. Update packages

```
root@control-plane:~# sudo apt update
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110
Hit:2 https://download.docker.com/linux/ubuntu jammy InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu jammy InRelease
Get:4 http://us.archive.ubuntu.com/ubuntu jammy-updates InRelease [11
Hit:5 http://us.archive.ubuntu.com/ubuntu jammy-backports InRelease
Fetched 229 kB in 3s (75.4 kB/s)
Reading package lists... 19%
```

2. Install required pacakges

```
root@control-plane:~# sudo apt install -y apt-transport-https curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
curl is already the newest version (7.81.0-1ubuntu1.15).
apt-transport-https is already the newest version (2.4.11).
0 upgraded, 0 newly installed, 0 to remove and 170 not upgraded.
root@control-plane:~#
```

3. Add the Kubernetes GPG key

```
root@control-plane:~# curl -s https://packages.cloud.google.com/apt/doc/apt-key
.gpg | sudo apt-key --keyring /usr/share/keyrings/kubernetes-archive-keyring.gp
g add -
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (
see apt-key(8)).
OK
root@control-plane:~#
```

4. Set the Kubernetes stable repository

```
root@control-plane:~# echo "deb [signed-by=/usr/share/keyrings/kubernetes-archi
ve-keyring.gpg] https://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee /
etc/apt/sources.list.d/kubernetes.list > /dev/null
root@control-plane:~#
```

5. Update the package list again

6. Install kubeadm, kubelet, and kubectl

```
root@control-plane:~# sudo apt install -y kubelet kubeadm kubectl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  conntrack cri-tools ebtables kubernetes-cni socat
The following NEW packages will be installed:
  conntrack cri-tools ebtables kubeadm kubectl kubelet kubernetes-cni socat
0 upgraded, 8 newly installed, 0 to remove and 170 not upgraded.
Need to get 87.1 MB of archives.
After this operation, 336 MB of additional disk space will be used.
Get:2 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 conntrack amd64 1:1.
4.6-2build2 [33.5 kB]
Get:6 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 ebtables amd64 2.0.1
1-4build2 [84.9 kB]
Get:8 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 socat amd64 1.7.4.1-
3ubuntu4 [349 kB]
5% [Connecting to packages.cloud.google.com] [8 socat 2,123 B/349 kB 1%]
```

R. INITIALIZE KUBERNETES

7. Hold on the versions to prevent automatic updates

```
root@control-plane:~# sudo apt-mark hold kubelet kubeadm kubectl
kubelet set on hold.
kubeadm set on hold.
kubectl set on hold.
root@control-plane:~#
```

8. Start and enable the kubectl services

```
root@control-plane:~# sudo systemctl enable kubelet
root@control-plane:~# sudo systemctl start kubelet
```

S. Install apt-transport-https, gnup and curl; for working with https repositories and downloading pacakges over the network.

```
fatima@control-plane:~$ sudo apt-get install -y apt-transport-https gnupg curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
curl is already the newest version (7.81.0-1ubuntu1.15).
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
apt-transport-https is already the newest version (2.4.11).
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
fatima@control-plane:~$
```

T. INSTALL MINIKUBE

```
root@control-plane: ~
root@control-plane:~# curl -LO https://storage.googleapis.com/minikube/releases
/latest/minikube-linux-amd64
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  9 89.3M    9 9038k    0     0  1221k      0  0:01:14  0:00:07  0:01:07 1871k
```

```
root@control-plane:~# sudo install minikube-linux-amd64 /usr/local/bin/minikube
root@control-plane:~#
```

U. Start Minikube

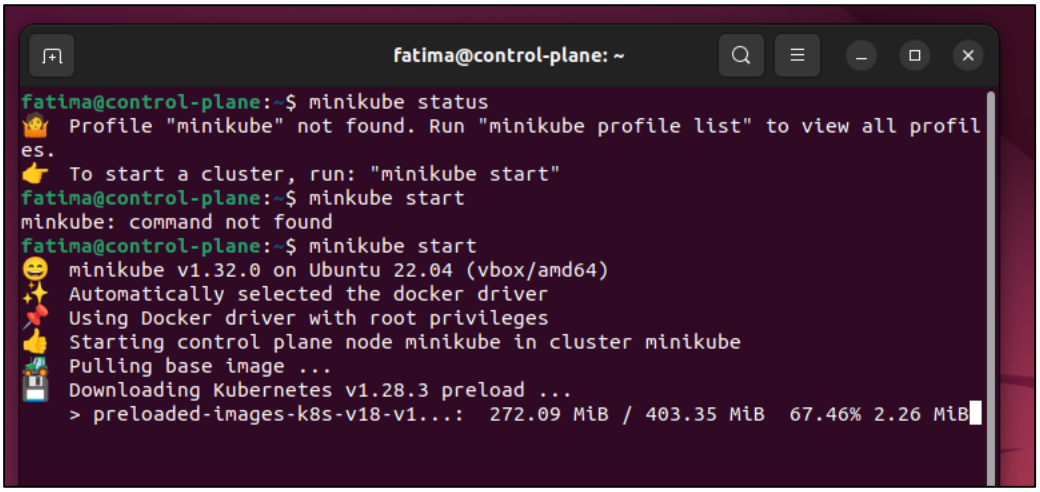
```
fatima@control-plane:~$ sudo minikube start --driver=docker --force
😄  minikube v1.32.0 on Ubuntu 22.04 (vbox/amd64)
❗  minikube skips various validations when --force is supplied; this may lead
to unexpected behavior
✨  Using the docker driver based on user configuration
🛑  The "docker" driver should not be used with root privileges. If you wish to
 continue as root, use --force.
💡  If you are running minikube within a VM, consider using --driver=none:
📘    https://minikube.sigs.k8s.io/docs/reference/drivers/none/
📌  Using Docker driver with root privileges
👍  Starting control plane node minikube in cluster minikube
🚜  Pulling base image ...
💾  Downloading Kubernetes v1.28.3 preload ...
```

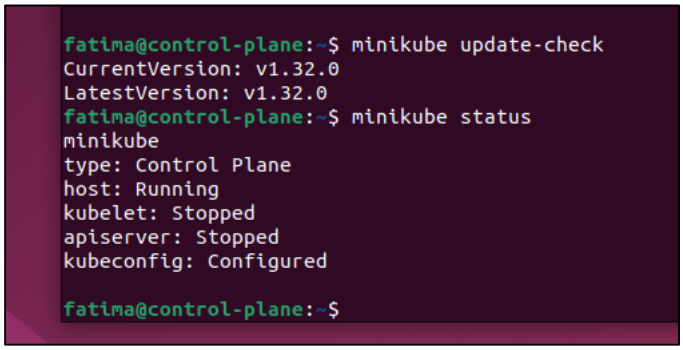
```
💾  Downloading Kubernetes v1.28.3 preload ...
    > gcr.io/k8s-minikube/kicbase...:  27.64 MiB / 453.90 MiB  6.09% 253.94 KiB
```

```
    > gcr.io/k8s-minikube/kicbase...:  453.90 MiB / 453.90 MiB  100.00% 935.90
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...\
```

```
fatima@control-plane:~$ sudo minikube start --driver=docker --force
😄  minikube v1.32.0 on Ubuntu 22.04 (vbox/amd64)
❗  minikube skips various validations when --force is supplied; this may lead to unexpect
ed behavior
✨  Using the docker driver based on user configuration
🛑  The "docker" driver should not be used with root privileges. If you wish to continue a
s root, use --force.
💡  If you are running minikube within a VM, consider using --driver=none:
📘    https://minikube.sigs.k8s.io/docs/reference/drivers/none/
📌  Using Docker driver with root privileges
👍  Starting control plane node minikube in cluster minikube
🚜  Pulling base image ...
💾  Downloading Kubernetes v1.28.3 preload ...
    > preloaded-images-k8s-v18-v1...:  403.35 MiB / 403.35 MiB  100.00% 1.90 Mi
    > gcr.io/k8s-minikube/kicbase...:  453.90 MiB / 453.90 MiB  100.00% 935.90
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...
🐳  Preparing Kubernetes v1.28.3 on Docker 24.0.7 ...
    ▪ Generating certificates and keys ...
    ▪ Booting up control plane ...
    ▪ Configuring RBAC rules ...
🔗  Configuring bridge CNI (Container Networking Interface) ...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🔎  Verifying Kubernetes components...
🌟  Enabled addons: default-storageclass, storage-provisioner
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by d
efault
```

```
fatima@control-plane:~$ minikube update-check
CurrentVersion: v1.32.0
LatestVersion: v1.32.0
fatima@control-plane:~$
```


fatima@control-plane: ~
fatima@control-plane:~$ minikube status
Profile "minikube" not found. Run "minikube profile list" to view all profiles.
To start a cluster, run: "minikube start"
fatima@control-plane:~$ minkube start
minkube: command not found
fatima@control-plane:~$ minikube start
minikube v1.32.0 on Ubuntu 22.04 (vbox/amd64)
Automatically selected the docker driver
Using Docker driver with root privileges
Starting control plane node minikube in cluster minikube
Pulling base image ...
Downloading Kubernetes v1.28.3 preload ...
> preloaded-images-k8s-v18-v1...: 272.09 MiB / 403.35 MiB 67.46% 2.26 MiB


fatima@control-plane:~$ minikube update-check
CurrentVersion: v1.32.0
LatestVersion: v1.32.0
fatima@control-plane:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Stopped
apiserver: Stopped
kubeconfig: Configured
fatima@control-plane:~$